

Session Code: DAT402

data systems

Building Database Applications With SQL Server “Yukon”

Istvan Cseri, Architect
Arpan Desai, Program Manager
Microsoft Corporation

istvanc@microsoft.com

arpande@microsoft.com

PDC⁰³

Make the connection

Microsoft[®]

WinFX Developer Preview

Tools

Visual Studio .net
Visual Basic .net
Visual C# .net
Visual J# .net
Visual C++ .net

Client Application Model

Avalon

System.Windows

Windows Forms

System.Windows.Forms

Web & Service Application Model

ASP.NET / Indigo

System.Web

Data Systems Application Model

Win FS

System.Storage

Yukon

System.Data.SqlServer

Mobile PC & Devices Application Model

Compact Framework

System.Windows.Forms

Mobile PC Optimized

System.Windows

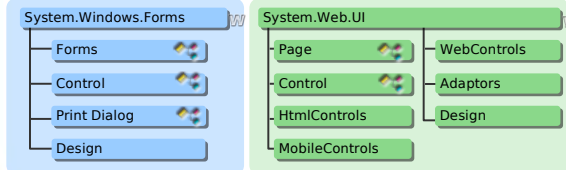
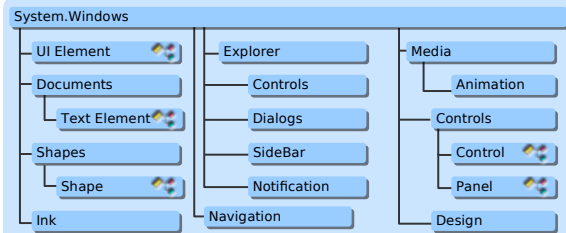
Command Line

System.Console

NT Service

System.ServiceProcess

Presentation



System.Help

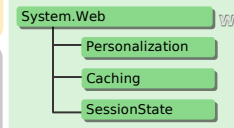
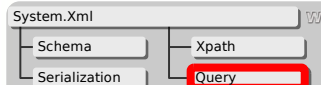
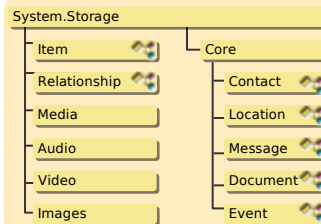
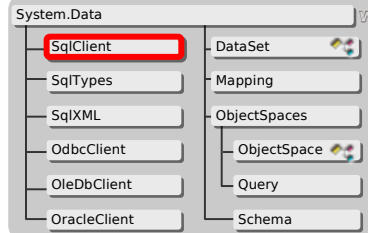
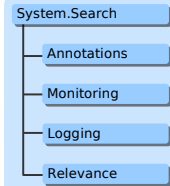
System.Drawing

System.NaturalLanguageServices

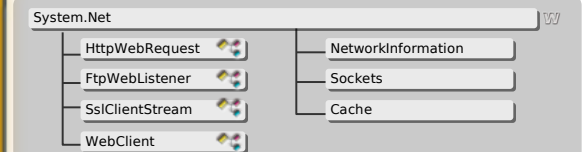
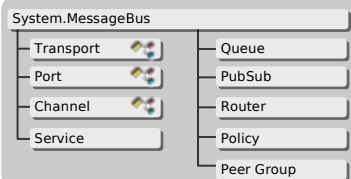
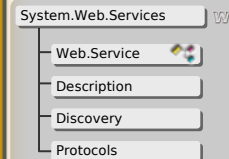
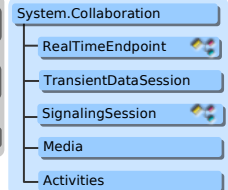
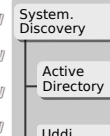
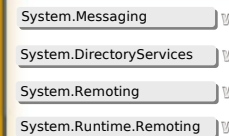
System.Speech

- Recognition
- Synthesis

Data

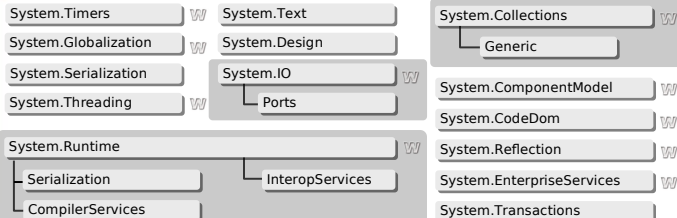


Communication

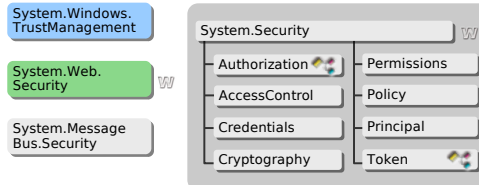


Fundamentals

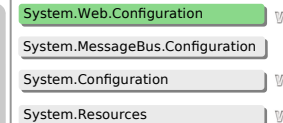
Base & Application Services



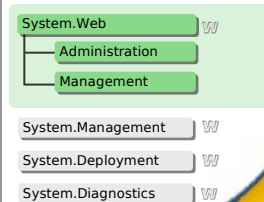
Security



Configuration



Deployment/Management



Agenda

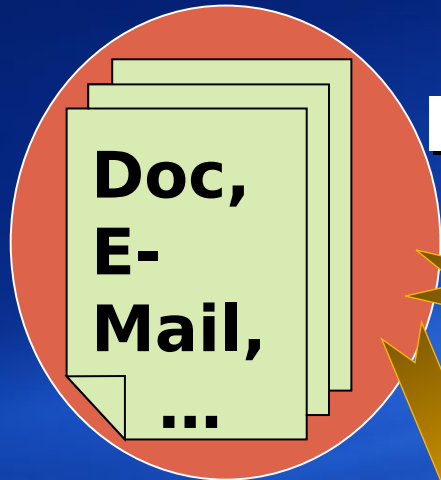
- SS2K Recap
- Scenarios
- Yukon XML support
 - Native XML storage
 - XML schema support
 - XML query with insert, update, and delete
- ADO.NET and Yukon
 - SQL Client support
 - XML Views
- Summary

SQL Server 2000 Technologies

- Server support
 - FOR XML – generate XML from tables
 - OpenXML – generate relational rowset from XML
- Mid-tier support (SQLXML)
 - XML views (annotated mapping schemas XSD)
 - Templates
 - UpdateGrams/BulkLoad
- Access methods
 - HTTP SOAP (via mid-tier ISAPI)
 - ADO, OLE DB; ADO.NET

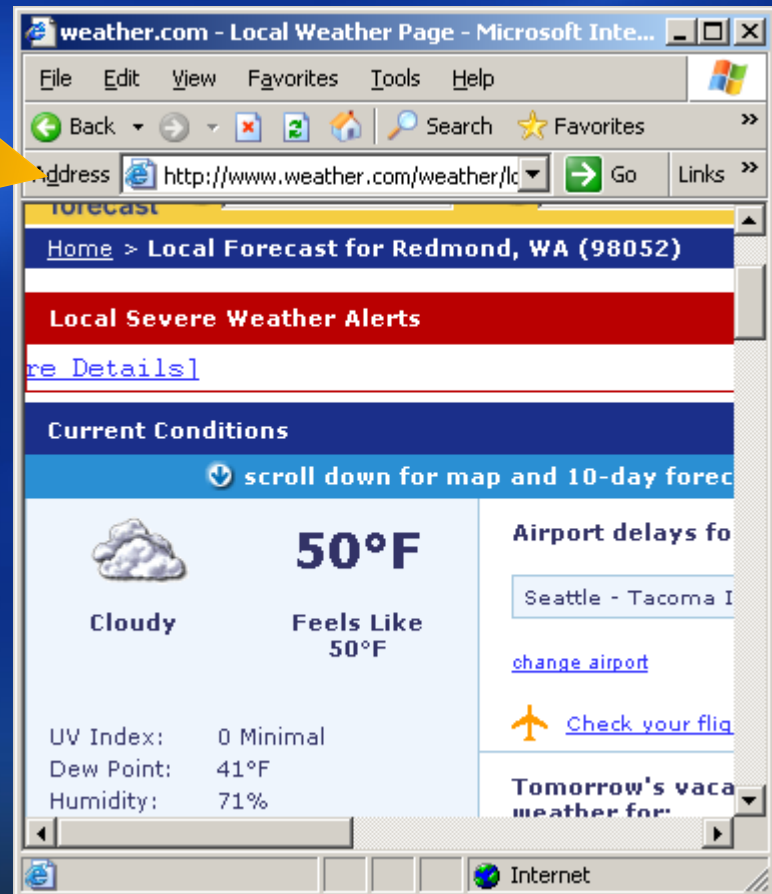
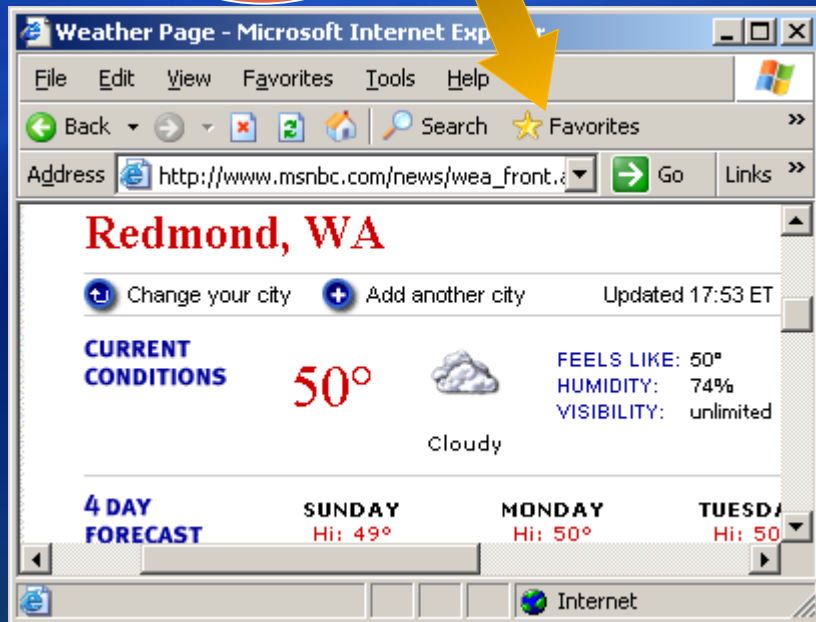
XML Scenarios

Document management...



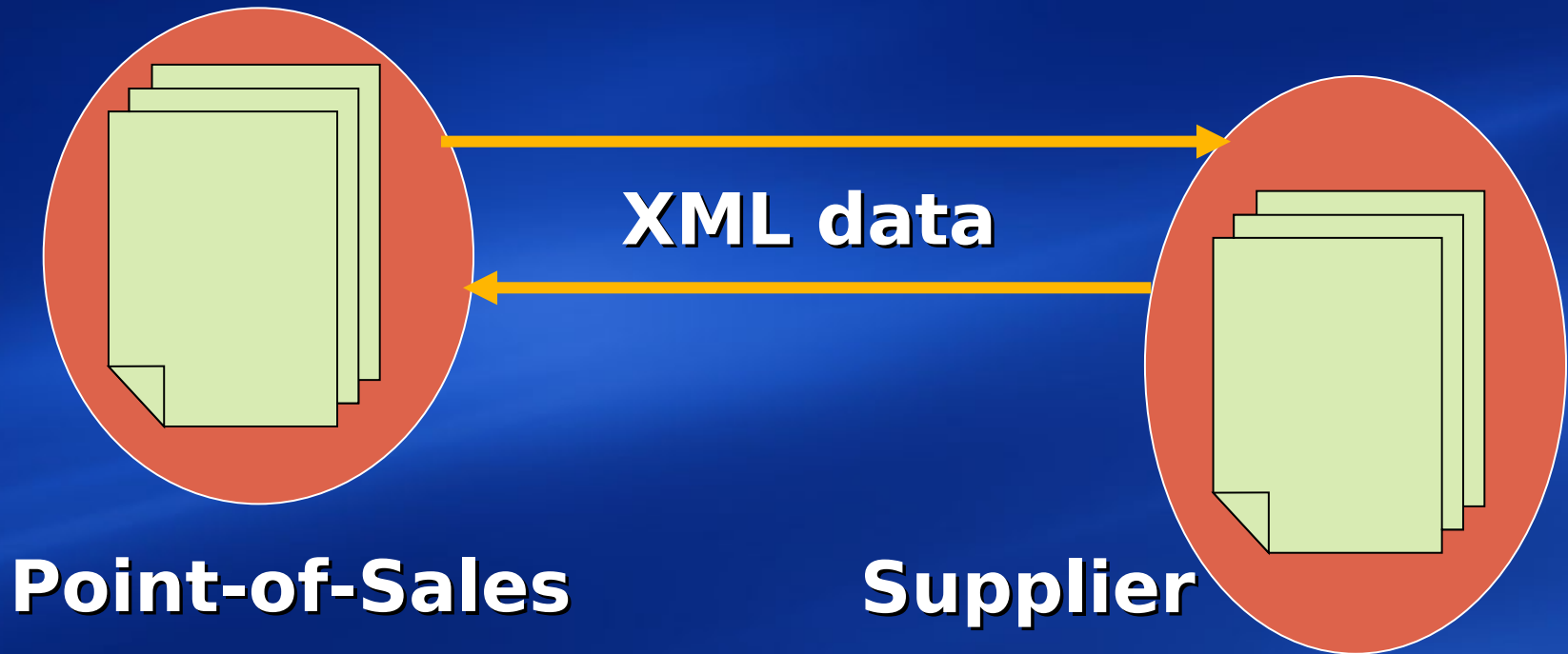
Index, search XML data

XSLT



XML Scenarios

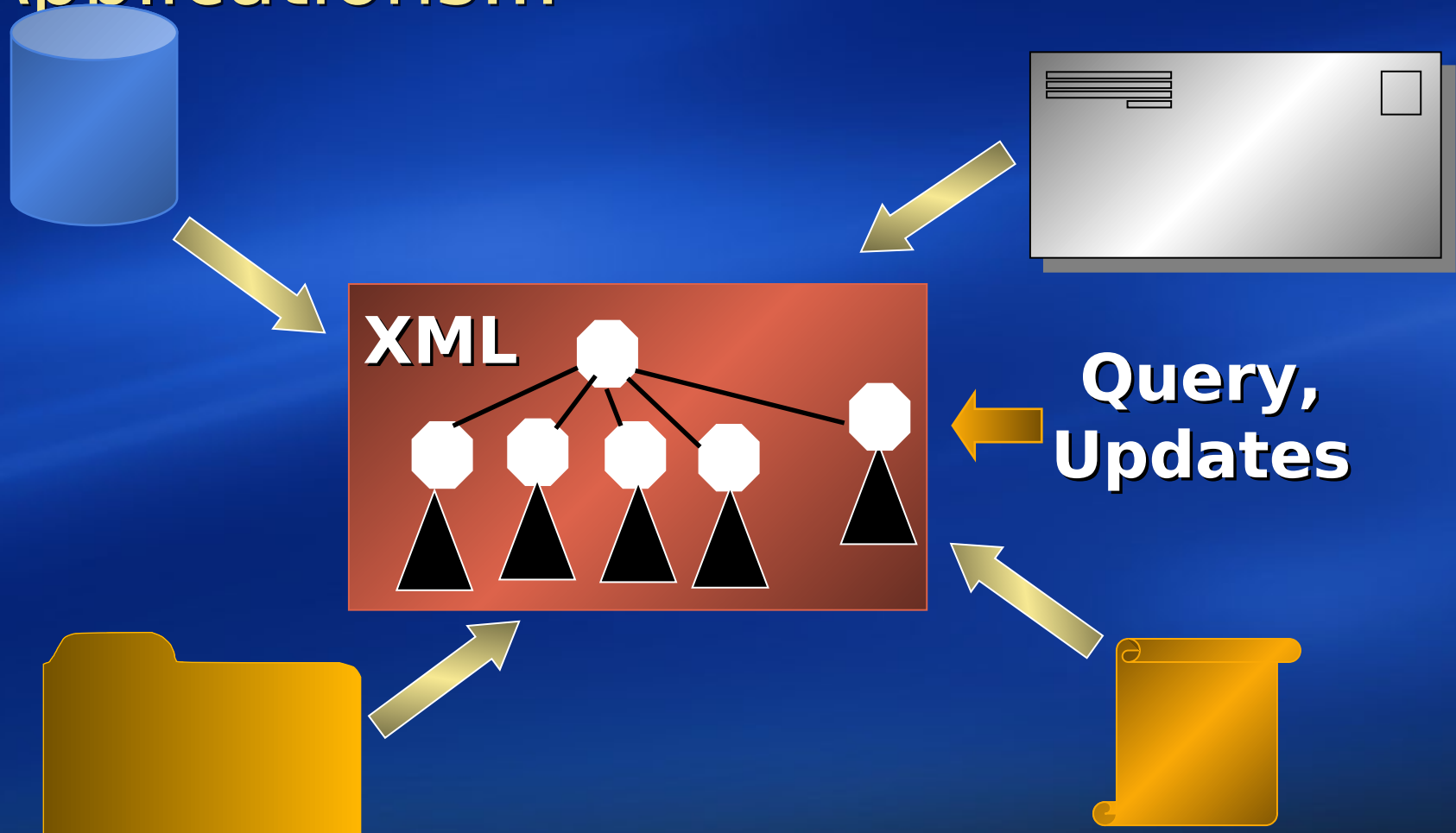
Data Exchange...



- Platform independent transport format
- Loosely-coupled system
- B2B, B2C, "A2A"

XML Scenarios

Mid-Tier Collaborative Applications...



Yukon XML Support

Relational and XML integration

- Structured and semi-structured data:
 - Structured data stored relationally
 - Bi-directional mapping between XML and relational data
 - Loosely structured and semi-structured data stored in native XML form
 - Index support with optimizer knowledge
- Deep SQL Server integration for XML
 - Triggers, replication, bulk load, security...

XML Support (Engine)

Highlights

- XML data type and XML index support
- Unified XML and relational store
 - Both SQL and XQuery supported by same industrial strength infrastructure
 - Leverages existing SQL engine and optimizer
- XQuery with data modification
- XML schema enforcement
- Client access using ADO.NET

FOR XML and OpenXML

- Fully backward compatible with SS2K
- FOR XML
 - New directive TYPE returns XML data type
 - Nested FOR XML
 - Assignment to XML data type
 - Support for new data types
- OpenXML:
 - XML overflow column
 - New types [n]varchar(max), varbinary(max), UDT

Native XML Store

XML Data Type

- Native SQL type
 - Use for column, variable or parameter
- ```
CREATE TABLE docs (id INT PRIMARY KEY,
 xDoc XML NOT NULL)
```
- Store un-typed or typed XML instances
  - Well-formed and validation checks
  - Methods on XML data type:
    - Query(), value(), exist(), XmlNodeRefs(), modify()
  - XML instances stored as LOB (2GB)
    - Efficient binary representation

# Full-Text Indexing

- XML filter
- Index and query XML instances
  - Markup tags removed
- Syntax — same as for other columns

```
CREATE FULLTEXT INDEX ON docs (xDoc)
```

- Use full-text search as filter, then XQuery search
  - Uses full-text index first
  - Uses XML index on tags, values, paths

# Full-Text Search

## Query examples

```
SELECT R.X::query ('//sec[@num=12]')
FROM
 (SELECT * FROM docs
 WHERE contains (xDoc, 'Wrd1 Wrd2'))
R(X)
```

- Supports xml:lang attribute

- ▣ Uses appropriate language word breaker

```
SELECT * FROM docs
WHERE contains (xDoc, 'Visionen',
 LANGUAGE 'German')
```

# XQuery and XML Datatype demo



# Native XML Store

## XML Index

- Create XML index on XML column

```
CREATE XML INDEX idx_1 ON docs (xDoc)
```

- Creates indexes on tags, values, paths
- Speeds up queries
  - Results can be served directly from index
  - Entire query is optimized
    - Same award winning cost based optimizer
  - Indexes are used as available

# XML Query

- XQuery: query XML documents and data
  - Standards-based: W3C working draft
- In document 123, return section heading of section 3 and later

```
SELECT id, xDoc::query('
 for $s in
 /doc[@id = 123]//sec[@num >= 3]
 return <topic>{data($s/heading)} </topic>
')
FROM docs
```

# XQuery Features

- FLWR: FOR / LET / WHERE / RETURN
- Includes XPath 2.0 (/doc[@id = 123])
- Element constructors (<topic>)
- Order-preserving operators
- Supports strong typing

# XML Data Modification

- Insert, update, and delete XQuery extensions
- XML sub-tree modification:
  - Add or delete XML sub-trees
  - Update values
- Add a new section after section 1:

```
UPDATE docs SET xDoc::modify('insert
<section num="2">
<heading>Background</heading>
</section>
after /doc/section[@num=1]')
```

# XML Schema Support

- XML Schema (W3C standard)
  - Rich mechanism for type definitions and validation constraints
  - Can be used to constrain XML documents
- Benefits of typed data
  - Guarantees shape of data
  - Allows storage and query optimizations
- XML type system
  - Store XML schemas in system meta-data

# XML Type System

- Associate XML namespace with XML type

- Query and modify typed XML data

```
DECLARE @x XML('http://www.ms.com/xdb')
SET @x = '<p:doc p="http://www.ms.com/xdb"...
>'

SELECT @x::query(
 p= "http://www.ms.com/xdb" 'namespace
/p:doc[@p:id = 123]//p:sec for $s in
<topic>{data($s/heading)}</topic>' return
FROM docs
```



# Strongly Typed XML Datatype

# demo

# Support for Yukon Inside ADO.NET

- Support within System.Data
  - SqlDataReader
    - `SqlXmlReader GetSqlXmlReader(int i);`
  - DataSet
    - New XPathDocument column type
    - Full databinding support
- Support within System.Xml
  - XML Views and XQuery

# XML Views

## 3-Part Mapping

- Schema-mapping separation
  - XML schema + relational schema + mapping
  - Optimizations resulting from having relational schema
- Relational domain extensibility
  - Support for SPs and UDFs
  - Inline SQL
- Tool Support

# XML Views

## Features

- XQuery support over the views
  - Optimized query translation into T-SQL queries
  - Leveraging MARS
  - Databinding over results
- BulkLoad - Bulk insert of XML data shredded into tables with ID propagation
- XmlAdapter - Instance based updates (XML equivalent of SqlDataAdapter)

# Using XML Views

# demo

# PDC<sup>03</sup>

Make the connection

**Microsoft Professional Developers Conference 2003**

October 26 - 30, 2003, Los Angeles, CA

**Microsoft®**